

***Remarks***

Reconsideration of this Application is respectfully requested.

Upon entry of the foregoing amendment, claims 1-19 are pending in the application. Claims 1, 3, 5, 8, 14, 16, and 18 are independent claims. Claim 5 is amended to define the claimed invention even more clearly. New claims 16 to 19 are sought to be added. These changes are believed to introduce no new matter, and their entry is respectfully requested.

Based on the above amendment and the following remarks, Applicants respectfully request that the Examiner reconsider all outstanding objections and rejections and that they be withdrawn.

The Abstract was objected for having an excessive length. Applicants submit herewith on a separate sheet a new Abstract having a length less than 150 words. The new Abstract is not intended to limit the present invention.

The drawings were objected for failing to provide a "Prior Art" label on FIGs. 1-4. Replacement sheets are attached hereto that include a "Prior Art" label on FIGs. 1-4.

Claims 1-8, 10, and 11 were rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Pat. No. 6,578,197, issued to Peercy *et al.* ("Peercy"), and U.S. Pat. No. 6,502,238, issued to Pavan *et al.* ("Pavan"). Applicants respectfully traverse.

Significant technical differences exist between the claimed invention and the applied references. Neither Peercy nor Pavan, taken alone or in combination, suggest or teach the claimed invention.

Peercy relates to translating procedural shading language instructions from one graphics API (such as Renderman) to another graphics API (such as OpenGL or

Direct3D). See, e.g., Peercy, col. 3, lns. 30-52 and col. 4, lns. 29-67. In FIG. 1, a graphics computation is expressed as a procedural shading expression (step 110, col. 6, lns. 12-14). The expression can be within a user application program in a high level language (such as "C") using procedures and functions of a shading language (such as Renderman). See, Peercy, col. 6, lns. 14-18. The application program is compiled (step 120) such that it is transformed to a sequence of parametric shading expressions which can be in a graphics language, such as, Open GL or Direct3D (step 130). See, Peercy, col. 6, lns. 21-35. This code is then targeted to a specific hardware platform (step 140) which produces executable code (step 150). In this way, high-level code, such as Renderman, can be targeted and sped up. See, Peercy, col. 6, lns. 35-43.

Pavan describes a distributed block-based programming model where programming blocks are distributed across a network of processing nodes. A user can specify interconnections between program blocks across the nodes. The system translates a user program to a system-level program and creates a program fragment for each node. See, e.g., Pavan, abstract, col. 2 lns. 17-51, and FIG. 1-2B. Program blocks can be logically characterized as source blocks that produce data, sink blocks that consume data and intermediate blocks that represent devices or software for processing functions. See, Pavan, col. 4, lns. 48-76. The program blocks can be interconnected through one or more ports. See, Pavan, col. 5, lns. 8-10.

Applicants respectfully submit this rejection based on a combination of Peercy and Pavan is improper. First, no motivation or suggestion to combine Peercy and Pavan is provided. Second, even if such a combination is assumed to be made for the sake of argument, Applicants submit that Peercy and Pavan do not teach each and every element of the claimed invention.

For instance, as admitted in the Office Action, Peercy does not teach or suggest storing processing blocks that define content and storing an application graph that expresses the identity of the stored processing blocks and data connectivity between the stored processing blocks as recited in steps (a) and (b) of claim 1. See, Office Action, p. 3. The tree used in Peercy and referenced by the Examiner is not an application graph stored for traversal during run-time as recited in claim 1.

Pavan further fails to overcome these deficiencies of Peercy. Pavan describes a distributed block-based programming model where programming blocks are distributed across a *network* of processing nodes. Pavan simply does not teach or suggest storing processing blocks that define content and storing an application graph that expresses the identity of the stored processing blocks and data connectivity between the stored processing blocks as recited in steps (a) and (b) of claim 1.

Independent claim 3 is further patentable over Peercy and Pavan, taken alone or in combination, for at least the same reasons as noted above.

Similarly, neither Peercy nor Pavan teach or suggest connections between blocks that implement data flow between blocks as in independent claims 5 and 8. For instance, independent claim 5 recites a game development and run-time system having *inter alia*:

a graphical application platform that enables a game application to run on any of multiple hardware platforms, said graphical application platform comprising:  
an application real-time kernel,  
a plurality of standard features implemented as executable blocks of logic, and  
connections between *said blocks that implement data flow between said blocks such that capabilities of the game application and any of the multiple hardware platforms can be implemented modularly* by adding additional corresponding blocks and connections. (emphasis added).

Likewise, claim 8 recites *inter alia*:

A graphical application platform for leveraging capabilities provided independently in at least one of an application software and a hardware platform, comprising:  
an application real-time kernel (ARK);  
a plurality of standard features implemented as executable blocks of logic; and  
connections between *said blocks that implement data flow between said blocks, whereby capabilities of at least one of the application software and the hardware platform can be implemented modularly* by adding additional corresponding blocks and connections.

Dependent claims 2, 4, 6-7, 10, and 11 are patentable for at least the same reasons as the independent claims from which they respectively depend and further in view of their own respective feature(s).

Claim 9 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Percy in view of Pavan as applied to claim 8, and further in view of U.S. Pat. No. 6,584,489, issued to Jones *et al.* ("Jones"). Applicants respectfully traverse.

Claim 9 is patentable over Percy and Pavan for at the same reasons provided above with respect to claim 8. Jones does not overcome the above-noted deficiencies and no motivation or suggestion to combine Jones with Percy or Pavan is provided. Further, Jones merely describes a method and system for scheduling use of a computer system resource. See, abstract. Jones does not teach or suggest a graphical application platform with an application real-time kernel including logic that invokes blocks according to a schedule as recited in claim 9.

Claim 12 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Percy in view of Pavan as applied to claim 8, and further in view of U.S. Pat. No. 5,857,106, issued to Barbour *et al.* ("Barbour"). Applicants respectfully traverse.

Claim 12 is patentable over Percy and Pavan for at the same reasons provided above with respect to claim 8. Barbour does not overcome the above-noted deficiencies

and no motivation or suggestion to combine Barbour with Percy or Pavan is provided. Barbour describes a library of generic and optimized modules (col. 1, lines 55-60 and FIG. 1). Different sets of optimized modules can be selected depending upon a particular processor/architecture configuration (FIG. 2, col. 3, lns. 14-40). Percy, Pavan and Barbour, taken alone or in combination, do not teach or suggest a graphical application platform with standard and additional blocks organized into components as recited in claim 12.

Claim 13 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Percy in view of Pavan in view of Barbour as applied to claim 12, and further in view of Jones. Applicants respectfully traverse and submit claim 13 is patentable for at least the same reasons provided above with respect to claim 12. Further, contrary to the assertion of the Examiner, Jones does not overcome the above-noted deficiencies and no motivation or suggestion to combine Jones with Percy, Pavan, or Barbour is provided. Applicants submit this combination of references is only arrived at based on impermissible hindsight.

Claims 14-15 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Percy in view of Barbour. Applicants respectfully traverse.

No motivation or suggestion to combine Percy and Barbour to arrive at the claimed invention is provided. Further, contrary to the assertion of the Examiner, neither Percy nor Barbour, taken alone or in combination, teach or suggest a method of pre-processing a graphics application with respect to a pre-defined hardware platform as recited in independent claim 14. Claim 14 recites *inter alia*:

a) selecting from among a set of alternative implementations of a feature;

- b) mapping at least one block, corresponding to the selected implementation, to a phase of execution;
- c) mapping the phase of execution to a stage of execution;
- d) creating a block execution order list corresponding to the stage of execution; and
- e) submitting the stage of execution to an application real time kernel for management of execution of the stage.

Dependent claim 15 is patentable for at least the same reason as independent claim 14 from which it depends and further in view of its own respective feature(s).

Claims 1-15 were further rejected under 35 U.S.C. § 103(a) as being unpatentable over Applicants' own allegedly admitted prior art. Applicants respectfully traverse. Each of claims 1-15 includes features not taught or suggested by pages 3-6 of Applicant's specification. The Examiner even lists some of the deficiencies in other approaches noted in pages 3-6 but *fails* to provide any specific basis or point to any admission by Applicant to support a rejection under 35 U.S.C. § 103(a). Applicants submit this is clearly improper as the Examiner has not set forth a *prima facie* case. If the Examiner insists upon maintaining this ground of rejection, Applicants request that a clear indication of support for this rejection be provided.

New claims 16-19 are also patentable.

### ***Conclusion***

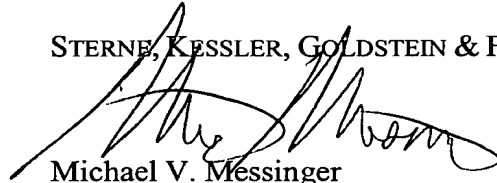
All of the stated grounds of objection and rejection have been properly traversed, accommodated, or rendered moot. Applicants therefore respectfully request that the Examiner reconsider all presently outstanding objections and rejections and that they be withdrawn. Applicants believe that a full and complete reply has been made to the outstanding Office Action and, as such, the present application is in condition for allowance. If the Examiner believes, for any reason, that personal communication will

expedite prosecution of this application, the Examiner is invited to telephone the undersigned at the number provided.

Prompt and favorable consideration of this Amendment and Reply is respectfully requested.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.

A handwritten signature in black ink, appearing to read "Michael V. Messinger", is written over the printed name and firm name.

Michael V. Messinger  
Attorney for Applicants  
Registration No. 37,575

Date: July 12, 2004

1100 New York Avenue, N.W.  
Washington, D.C. 20005-3934  
(202) 371-2600

286209